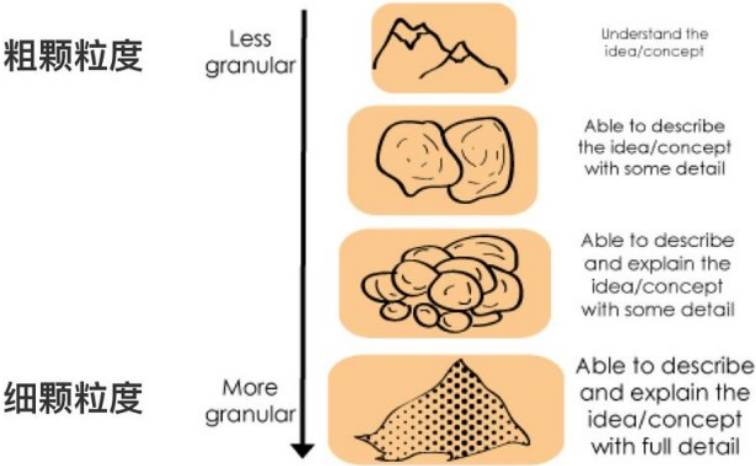


1. Granularity - 粒度

Granu · lar · ity = grain + clarity

The term “granularity” means working for or towards clarity. At the most foundational level, granularity = grain + clarity. In order to fully understand what this word means and all that it encompasses.



ref:
<https://sites.ualberta.ca/~obilash/granularity.html#:~:text=The%20term%20%E2%80%9Cgranularity%E2%80%9D%20means%20working%20for%20or%20towards,most%20foundational%20level%2C%20granularity%20%3D%20grain%20%2B%20clarity.>

2024 年网络流行语之职场黑话：
来源影片：《年会不能停！》

“对齐颗粒度”指通过核对工作细节使沟通双方形成共同认知的职场术语，演员大鹏在电影宣发中阐释其本质为“同步信息以避免理解偏差”。该词因 2024 年 1 月上映的电影《年会不能停！》中高频出现而成为现象级黑话。

ref:
<https://baike.baidu.com/item/%E5%AF%B9%E9%BD%90%E9%A2%97%E7%B2%92%E5%BA%A6/63909820>

回到数据标注：

粒度的本质是信息的抽象层次与细致程度。其使用在图像分割（Image Segmentation）、自然语言处理（NLP）、多模态（Multi-Modal）中均表达对不同细致程度的分筛和对应。

duìqíkēlìdù

对 齐 颗 粒 度

词汇释义

对齐-align 颗粒度-granularity

对齐颗粒度即**信息同步**，通常用于描述在数据处理、系统设计或项目管理中，如何调整或匹配不同元素的细致程度或精度。

强调的是在细节层面上的一致性和协调性。

对于图像分割而言：

粒度	描述	主要任务
1. 图像级 Image	为整张图像分配一个或多个全局标签	图像分类、多标签分类
2. 目标级 Category	标识出图像中每个感兴趣物体的空间位置和类别	目标检测、姿态估计
2.1 边界框 Sub-category	用矩形框近似物体的位置和范围	目标检测
2.2 关键点 Key point	标注物体的一系列预定义特征点	人体姿态估计、人脸关键点检测
3. 像素级 pixel	为图像中的每一个像素分配一个语义类别或实例标识	图像分割
3.1 语义分割 Semantic	每个像素被分类为某个语义类别，不区分个体	场景解析
3.2 实例分割 Instance	区分不同物体的个体，即使它们属于同一类别	实例分割
3.3 全景分割 Panoptic	统一语义分割和实例分割，为所有像素分配唯一的类别和实例 ID，包括“事物”（可数的，如人、车）和“背景”（不可数的，如天空、草地）	全景分割

Further Info : <https://zhuanlan.zhihu.com/p/368904941>

对于语言模型，粒度用于描述分词算法的精细度：

粒度	词 Word	子词 Sub-Word	字 Char (Character)
常用算法	隐式马尔科夫模型 HMM	BPE, WordPiece...	Char-Based RNN
特征	语义单元明确，对于成熟语言或词典完备的场景效果稳定	平衡了词表大小与语义，具有极好的泛化能力	完全避免了分词错误，但序列长度急剧增加，难以直接捕获词语级别的语义

Further Info: <https://zhuanlan.zhihu.com/p/50444885>

2. Learning Paradigm- 学习范式

1. 监督学习(Supervised Learning)

定义：在有标签 (label)的数据集上训练模型，使其能够学会输入到输出的映射关系。

常见任务：

分类(Classification): 图像分类(ResNet, ViT)、文本分类(BERT, GPT 分类头)、语音识别

回归(Regression): 房价预测、目标检测中的边界框回归(YOLO, Faster RCNN)

序列预测(Sequence Prediction): 时间序列预测, 语音识别(CTC,Transformer ASR)

2. 无监督学习(Unsupervised Learning)

定义：模型在无标签数据上训练，发现数据的潜在结构或模式。

常见任务：

聚类(Clustering): K-Means, DBSCAN, GMM

降维(Dimensionality Reduction): PCA, t-SNE, UMAP

表示学习(Representation Learning): Autoencoder(AE, VAE), Word2Vec, Contrastive Learning (SimCLR, MoCo)

3. 半监督学习(Semi-Supervised Learning)

定义：利用少量标注数据 + 大量未标注数据进行训练，结合监督学习和无监督学习的优点。

常见方法：

一致性正则化(Consistency Regularization): MixMatch, FixMatch, Noisy Student

伪标签(Pseudo-Labeling): Pseudo-Label, Self-Training

对比学习(Contrastive Learning): SimCLR, MoCo + 有监督任务

4. 生成式学习 (Generative Learning)

定义：学习数据的分布，以生成新的、逼真的数据样本。

常见任务：

图像生成: GAN (StyleGAN, BigGAN), Diffusion Model (Stable Diffusion, Imagen)

文本生成: GPT, T5, Llama

语音生成: WaveNet, VITS

视频生成: Make-A-Video, Video Diffusion

5. 自监督学习(Self-Supervised Learning)

定义：从数据本身生成伪标签，通过构造辅助任务 (pretext tasks)进行学习。

常见任务：

对比学习(Contrastive Learning): SimCLR, MoCo, BYOL, DINO

掩码预测(Masked Prediction) :BERT, MAE(Masked Autoencoder)

旋转预测(Rotation Prediction): 让模型预测图片旋转角度

6. 强化学习(Reinforcement Learning, RL)

定义：智能体(Agent)在环境(Environment)中与环境交互，通过奖励机制(Reward) 优化策略 (Policy)，最大化长期回报(Return)。

常见任务：

游戏: AlphaGo, AlphaStar, MuZero（围棋、StarCraft、Atari）

机器人控制: 四足机器人、机械臂

自动驾驶: Tesla FSD（部分基于 RL）

交易策略: 量化金融

学习范式	目标	典型模型	主要应用
监督学习	学习输入到输出的映射关系	CNN, Transformer	图像分类、语音识别
无监督学习	发现数据的隐藏模式	K-Means, Autoencoder	聚类、表示学习
半监督学习	利用未标注数据中的内在结构提升模型泛化能力	FixMatch, Pseudo-Label	医疗、NLP
生成式学习	生成新数据	GAN, Diffusion, GPT	图像/文本/语音生成
自监督学习	自生成伪标签进行学习	BERT, SimCLR, MAE	预训练大模型
强化学习	通过奖励优化策略	DQN, PPO, MuZero	机器人、游戏 AI

ref: https://blog.csdn.net/m0_52654914/article/details/146987832

监督学习的子类学习方法

1. 强监督学习

定义：使用高质量、高精度、与任务目标完全一致的标注数据进行训练。

特点：监督信号是直接、干净、无噪声的。

示例：

- 为了训练一个猫狗分类器，我们为每张图片提供了精确的“猫”或“狗”的标签。（图像级，精确）
- 为了训练一个肿瘤分割模型，我们由专业放射科医生绘制了像素级的分割掩码。（像素级，精确且直接）

对模型的影响：

优点：提供了最清晰的优化目标，通常能引导模型达到当前架构下的最高性能上限。

缺点：获取此类标注的成本极高，且严重依赖标注质量。

2. 弱监督学习

这是一个总称，指使用质量较低、较粗糙或不完全准确的标注进行训练。根据标注“不完美”的类型，可细分为三类（此分类取自《机器学习》-周志华）：

a. 不完全监督

定义：只有训练集的一个子集被标注，其余数据则无标注。

本质：这其实就是半监督学习。它属于弱监督的一个子类，因为模型被迫从不完整的监督信号中学习。

示例：在 1 万张图像中，只有 100 张有类别标签，其余 9900 张没有。

b. 不精确监督

定义：我们拥有的标注是粗粒度的，不如我们希望模型做出的预测那样精确。

本质：“粒度”概念的直接体现，是弱监督中最常见和研究最广泛的一类。

示例：

- 任务目标：像素级语义分割。
提供的标注：图像级标签（只有“图中有猫”）。
挑战：模型需要从“整张图有猫”这个粗粒度信号，去学习“哪个像素是猫”这个细粒度任务。

c. 不准确监督

定义：标注中含有错误，即标注本身不完全正确。

本质：标签噪声问题。

示例：在猫狗分类的数据集中，部分“猫”的图片被错误地标成了“狗”。

3. 损失函数 (Loss Function)

损失函数是一个将决策或预测结果映射到一个实数值的函数，该数值表示在给定的输入数据下，该决策或预测所带来的成本或误差。

对于一个数据点，假设其真实值为 y ，模型的预测值为 \hat{y} ，则损失函数 L 定义为：

$$L(y, \hat{y})$$

该函数输出一个非负的标量值。损失值越大，表示预测越不准确。

目标：机器学习模型的目标是通过调整其参数 θ ，在训练数据集上最小化这个损失函数的平均值（即经验风险）。

根据任务类型（回归、分类、生成等），损失函数的选择亦不同。

1. 回归任务

回归任务的目标是预测一个连续值。

均方误差：

$$L(y, \hat{y}) = (y - \hat{y})^2$$

平均均方误差 (MSE)：

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

特性：

对离群点非常敏感，因为误差被平方放大。

具有连续且光滑的导数，利于梯度下降等优化算法。

从概率视角看，最小化 MSE 等价于在假设噪声服从高斯分布下进行最大似然估计。

绝对误差：

$$L(y, \hat{y}) = |y - \hat{y}|$$

平均绝对误差 (MAE)：

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

特性：

对离群点不如 MSE 敏感，更加稳健。

在 $y = \hat{y}$ 处导数不存在（不可微），优化可能更复杂。

Huber Loss

$$L_{\delta} = \begin{cases} \frac{1}{2}(y - \hat{y})^2, & \text{for } |y - \hat{y}| \leq \delta \\ \delta|y - \hat{y}| - \frac{1}{2}\delta^2, & \text{otherwise} \end{cases}$$

特性：

结合了 MSE 和 MAE 的优点。在误差较小时，它像 MSE 一样具有光滑的导数；在误差较大时，它像 MAE 一样，对离群点不敏感。

δ 是一个超参数，用于控制从二次行为切换到线性行为的阈值。

2. 分类任务

分类任务的目标是预测一个离散的类别标签。

交叉熵损失(Cross-Entropy Loss)

二分类交叉熵 (Binary Cross-Entropy):

真实标签 ($y \in \{0,1\}$), 预测概率 ($\hat{y} = P(y = 1)$)。

$$L(y, \hat{y}) = -[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})]$$

多分类交叉熵(Categorical Cross-Entropy): 真实标签通常表示为 **one-hot** 向量 y (例如, $([0,0,1,0])$) , 模型输出一个概率分布 \hat{y} 。

$$L(y, \hat{y}) = - \sum_{c=1}^C y_c \log(\hat{y}_c)$$

其中 C 是类别总数。因为 y 是 **one-hot** 向量, 实际上只有真实类别 c 对应的项 $\log(\hat{y}_c)$ 对损失有贡献。

损失函数与模型训练

1. 优化算法(Optimization Algorithm)

定义了损失函数后, 我们需要一个算法来寻找使其最小化的模型参数 θ 。最常用的方法是**梯度下降(Gradient Descent)** 及其变种 (如随机梯度下降 **Stochastic Gradient Descent (SGD)**, **Adam**)。

->计算损失函数关于模型参数的梯度 $\nabla_{\theta} J(\theta)$, 然后沿着梯度的反方向更新参数, 逐步逼近最小值点。

Regularization 正则化

为了防止过拟合, 我们通常在损失函数中加入一个正则化项 (**Regularization Term**), 构成新的损失函数。

- **L2 正则化:** 在原始损失 $J(\theta)$ 上加上权重的平方和 (**L2 范数**), 即

$$J_{total}(\theta) = J(\theta) + \lambda \sum \theta_i^2$$

这倾向于让权重变小, 趋于平滑。

- **L1 正则化:** 加上权重的绝对值之和 (**L1 范数**), 即

$$J_{total}(\theta) = J(\theta) + \lambda \sum |\theta_i|$$

这倾向于产生稀疏的权重矩阵, 即部分权重为 0, 可用于特征选择。

4. 归纳偏置(Inductive Biase) / 学习偏置 (Learning Biase)

No free lunch (NFL) theorem for Optimization

For certain types of mathematical problems, the computational cost of finding a solution, averaged over all problems in the class, is the same for any solution method.

-> No method offers a "short cut"

->对于任意函数而言，一定的偏好（归纳偏置）是实现泛化的必要手段，i.e., 没有完全通用的学习算法，任何学习算法都只能在特定的一些分布上实现泛化。

Inductive Bias	Corresponding property
Distributed representations	Inputs mapped to patterns of features
Convolution	group equivariance (usually over space)
Deep architectures	Complicated functions = composition of simpler ones
Graph Neural Networks	equivariance over entities and relations
Recurrent Nets	equivariance over time
Soft attention	equivariance over permutations
Self-supervised pre-training	$P(X)$ is informative about $P(Y X)$

Table 1: Examples of current inductive biases in deep learning. Many have to do with the architecture while the last one influences the training framework and objective.

Inductive biases for deep learning of higher-level cognition

<https://arxiv.org/pdf/2011.15091>

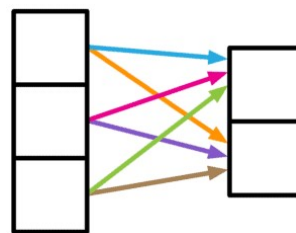
归纳偏差：学习算法中为引导模型趋向于某些特定解而内置的一组假设和先验知识。在数据有限的情况下，归纳偏差能有效提升性能；但数据量充足时，基线模型表现更优。

-> 在模型算法里反映假定的数据集分布。

关系归纳偏差(Relational inductive biases)

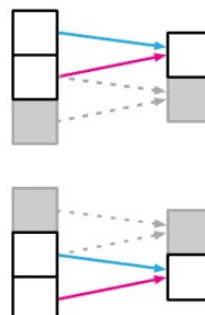
Weak Relation :

Relationship between the neural units is weak, meaning that they're somewhat independent of each other. A fully connected layer in the net can represent this kind of relationship:



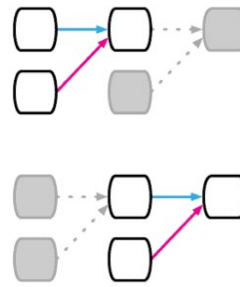
Locality:

To capture an image's local information. One way to do that is the use of a convolutional layer. It can capture the local relationship between the pixels of an image. Then, as we go deeper in the model, the local feature extractors help to extract the global features:



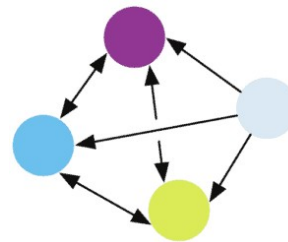
Sequential Relation:

Sometimes our data has a sequential characteristic. For instance, time series and sentences consist of sequential elements that appear one after another. To model this pattern, we can introduce a recurrent layer to our network:



Arbitrary Relation:

To solve problems related to a group of things or people, it might be more informative to see them as a graph. The graph structure imposes arbitrary relationships between the entities, which is ideal when there's no clear sequential or local relation in the model:



Ref: <https://www.baeldung.com/cs/ml-inductive-bias>

非关系归纳偏差(Non-relational inductive biases)

1. 非线性激活函数 (Non-linear Activation Function)

允许模型捕获隐藏在数据中的非线性。深度神经网络的核心，多个纯线性层的组合仍然是线性层，将让网络退化为上述关系归纳偏差。

2. Dropout

一种正则化技术，通过强制网络的随机子集分别学习数据模式来帮助网络避免直接记忆训练数据。获得的模型最终能够更好地泛化并避免过拟合。

3. Weight Decay (权重衰减)

另一种对模型权重施加约束的正则化方法。权重衰减有多种版本，常见 L1 和 L2 正则化技术。权重衰减不会让权重变得非常大，从而防止模型过度拟合。

4. Normalization (归一化)

加快训练速度和正则化，减少了网络激活分布的变化，也就是内部协变量偏移 (Internal co-variate shift)。

5. Data Augmentation (数据增强)

可以将数据增强视为另一种正则化方法。它对模型施加的内容取决于其算法。例如，在句子中添加噪音或单词替换是两种类型的数据增强。他们假设添加噪音或单词替换不应更改分类任务中单词序列的类别。

6. Optimization Algorithm (优化算法)

对训练结果的迭代优化算法。例如不同版本的梯度下降算法，其可以导致不同的最优值。随后，生成的模型将具有其他泛化属性。此外，每种优化算法都有自己的参数，可以极大地影响模型的收敛性和最优性。

5. 抽象层级(Level of Abstraction)

抽象层级是一个系统性的框架，用于在描述、分析或实现一个复杂系统时，设定信息细节的边界和范畴。它不是一个物理属性，而是一种认知约定(Cognitive Convention)或工程规范(Engineering Discipline)，旨在通过选择性忽略或封装下层系统的特定细节，来管理系统的复杂性。

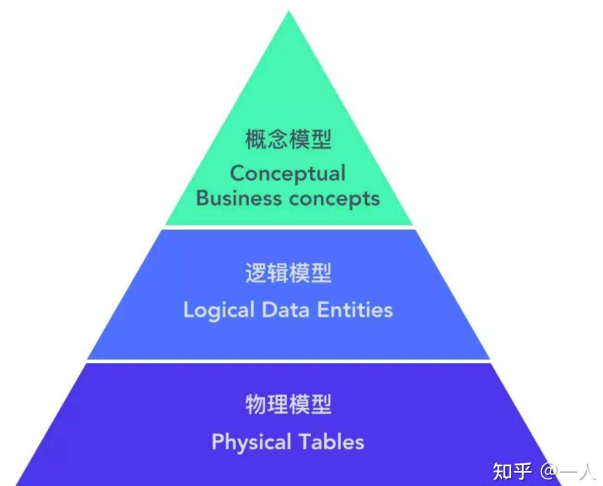
即任何系统都可以被分解为多个层次，每一层次为其上层提供服务(Services) 或功能(Functionality)，同时将其下层的实现细节(Implementation Details) 隐藏起来。这种隐藏被称为信息隐藏(Information Hiding) 或封装(Encapsulation)。

抽象层级的核心特征

1. 信息内容的粒度(Granularity of Information):

高抽象层级(High Level of Abstraction): 信息被高度聚合，使用接近问题域(Problem Domain)的词汇进行描述。描述的是“做什么(What)”，而非“如何做(How)”。例如，“渲染用户界面”或“处理支付交易”。

低抽象层级(Low Level of Abstraction): 信息粒度极细，使用接近解决方案域(Solution Domain)或物理实现的词汇。描述的是具体的操作、数据结构和状态变化。例如，“将寄存器 AX 的值移动到内存地址 0xFFFF”或“分配 64 字节内存”。



<https://zhuanlan.zhihu.com/p/29289790848>

2. 规范性与描述性(Specification vs. Description):

高层抽象通常是规范性(Prescriptive)的, 它定义了系统的目标、行为和接口契约。

低层抽象通常是描述性(Descriptive)的, 它精确地说明了行为是如何通过一系列具体步骤实现的。

3. 认知负荷与可控性(Cognitive Load vs. Control):

高层抽象通过隐藏复杂性, 显著降低了理解和使用系统所需的认知负荷(Cognitive Load)。

低层抽象则提供了对系统更精细、更直接的控制能力, 但代价是极大地增加了认知负荷和出错的可能性。

抽象层级的运用依赖于两个互逆的过程:

抽象(Abstraction): 这是一个向上(Upward)的过程。通过识别并提取系统中的基本特征, 同时忽略与当前目标无关的细节, 从而形成一个更高层级的模型。

精化(Refinement): 这是一个向下(Downward)的过程。它将高层级的抽象描述, 逐步转化为更具体、更详细的低层级描述, 直至达到可实现的级别。在形式化方法(Formal Methods)中, 精化是一个严格的数学过程, 用以证明低层设计正确实现了高层规范。

4. 形式化描述与选择标准

一个抽象层级可以形式化地定义为一个系统模型(System Model), 该模型包含:

- 一组相关的状态(States)。
- 一组在这些状态上定义的操作(Operations) 或行为(Behaviors)。
- 一个抽象函数(Abstraction Function), 用于将低层、更具体的状态映射到高层、更抽象的状态。

选择合适的抽象层级取决于分析目标(Analysis Goal):

系统级设计与沟通: 需要高抽象层级。

性能优化与错误诊断: 需要深入到低抽象层级。

接口定义与模块化: 需要在相邻抽象层级之间定义清晰、稳定的契约(Contract)。

6. 标签噪声与歧义 Label Noise & Ambiguity

标签噪声(Label Noise)

定义：标签噪声指的是训练数据集中标签(label)存在的错误或不准确性。这些错误可能源于人工标注、数据收集或处理过程，导致标签与真实值(ground truth)不一致。

随机噪声(Random Noise): 标签错误随机发生，例如由于标注者的疏忽或随机误差。

系统噪声(Systematic Noise): 标签错误由系统性偏差引起，例如标注规则不统一或工具故障。

类依赖噪声(Class-Dependent Noise): 某些类别(class)的标签更容易出错，通常与类别不平衡或标注难度相关。

原因：

人工标注错误(human annotation error)，如疲劳或误解。

数据采集缺陷(data collection flaws)，如传感器噪声或传输错误。

标注协议不明确(ambiguous annotation protocols)，导致不一致的标签。

->模型可能过拟合(overfitting)噪声，降低泛化能力(generalization)。

训练过程不稳定，增加收敛时间。

性能指标（如准确率）下降，尤其在复杂任务中。

歧义(Ambiguity)

定义：歧义指的是数据或标签本身固有的不明确性，使得样本难以被清晰分类或解释。与标签噪声不同，歧义并非错误，而是源于数据的内在特性。

类别歧义(Class Ambiguity): 一个样本可能同时属于多个类别，例如在图像分类中，物体可能介于两个类别之间（如“猫”和“狗”的混合）。

特征歧义(Feature Ambiguity): 数据的特征表示不清晰或重叠，导致决策边界模糊，例如在自然语言处理中，词语的多义性。

标注歧义(Annotation Ambiguity): 标注过程中，由于主观性或不完整信息，标签本身存在多种合理解释。

原因：

数据复杂性(data complexity)，如真实世界场景的多变性。

标注标准主观(subjective annotation criteria)，不同标注者可能有不同判断。

信息不足(insufficient information)，例如低分辨率图像或简短文本。

->模型不确定性(uncertainty)增加，输出概率分布更平坦。

决策边界不清晰，可能导致模型混淆。

在评估时，传统指标（如准确率）可能无法充分反映模型性能。

7. 类别不平衡 Class Imbalance

定义：类别不平衡是指在分类问题中，类别标签的分布高度不均匀，即某些类别（多数类，**Majority Class**）的样本数量远多于其他类别（少数类，**Minority Class**）。这常见于现实应用，如欺诈检测、医疗诊断，其中正例（**Positive Class**）往往只占小部分。

自然现象：真实世界中某些事件罕见（如疾病爆发）。

数据收集偏差：采样方法导致某些类别过代表或欠代表。

模型偏差(**Model Bias**):模型可能过度拟合多数类，忽略少数类，导致准确率(**Accuracy**)虚高但少数类识别率低。=

评估指标误导：准确率不再可靠，需使用针对性指标，如精确率(**Precision**)、召回率(**Recall**)、F1 分数(**F1-Score**)或 AUC-ROC(**Area Under the Receiver Operating Characteristic Curve**)。

训练不稳定：梯度下降(**Gradient Descent**)等优化过程可能收敛到次优解。

数据层面：

过采样(**Oversampling**): 增加少数类样本，如 **SMOTE**(**Synthetic Minority Over-sampling Technique**)。

欠采样(**Undersampling**): 减少多数类样本，可能丢失信息。

算法层面：

代价敏感学习(**Cost-Sensitive Learning**): 为不同类别分配不同误分类代价。

集成方法(**Ensemble Methods**): 如平衡随机森林(**Balanced Random Forest**)。

评估层面：

使用混淆矩阵(**Confusion Matrix**)和平衡准确率(**Balanced Accuracy**)。